MULTIMEDIA UNIVERSITY

STUDENT ID NO

MULTIMEDIA UNIVERSITY

FINAL EXAMINATION

TRIMESTER 1, 2016/2017

TOP2121 – OBJECT-ORIENTED PROGRAMMING

(All sections / Groups)

13 OCTOBER 2016
9.00a.m –11.00a.m
(2 Hours)

INSTRUCTIONS TO STUDENTS

1. This question paper consists of 6 pages, including the cover page, with **FOUR** questions only.

2. Attempt **ALL FOUR** questions. All questions carry equal marks and the distribution of the marks for each question is given.

3. Please print all your answers in the answer booklet provided.

Question 1

(a) Object-oriented programming uses the concept of *class*. List **FOUR** properties of *class*.

[4 marks]

(b) There are **FOUR** syntaxes in the Java code below. State the meaning of each syntax:

```
public static void passOOP();
```

[4 marks]

(c) Describe **TWO** advantages of object-oriented programming compared to procedural programming?

[2 marks]

Question 2

(a) Write down the output for the Java codes as shown in Figure 1:

```
public class A{
   int i, j;

  A(int i){
  this.i=i;
  System.out.println ("Inside A (int "+i+")");   }

   A(int i, int j){
     this.i=i;
     this.j=j;
     System.out.println ("Inside A (int "+i+", int "+j+")");   }
     }

class B extends A {
     int k;

B(int i, int j, int k){
     super(i,j);
     this.k=k;
     System.out.println ("Inside B (int i, int j, int k)");  }

public static void main(String[] args){
       B obj = new B(2,3,4);  }
}
```

Figure 1. A Sample Java Code.

[4 marks]

Continued...

(b) There are **TWO** methods of applying multithreading in Java, namely, extending Thread or implementing Runnable interface. If the `class B` in Figure 1 is applying multithreading, which method is more appropriate? Explain your answer.

[2 marks]

(c) Access Modifier in Java allows programmer to control the access to object's variables and methods in different conditions. Construct the table below by applying *Yes* or *No* to indicate the accessibility in the conditions *Subclasses* and *Same Package*.

Condition / Modifier	Same Class	Subclasses	Same Package	Different Package Classes
Public	Y			Y
Protected	Y			N
Default	Y			N
Private	Y			N

[4 marks]

Question 3

(a) Draw a UML class diagram based on the java codes given below. Show the attributes and methods of the class including modifiers.

```
package HR;
public class Employee {

  private String name;
  private double pay;
  private final int EMPLOYEE_ID;

  public static final double STARTING_PAY_RATE = … ;

  public Employee(String name, double pay) {
         …
  }
  public String getName() { … }
  public int getEmployeeID() { … }
  public double getPay() { … }
  public void changeName(String newName) { … }
  public void changePay(double newPay) { … }
}
```

[4 marks]

Continued...

(b) Given the scenario as follows, define the necessary classes and draw a relationship diagram for the classes:

- FIST has students who register subjects.
- FIST differentiates new students from repeat students, since they are billed differently.

[3 marks]

(c) Draw a state diagram for the Java Applet used for login as shown in Figure 2. The same login dialogue will be displayed if the login attempt fails.

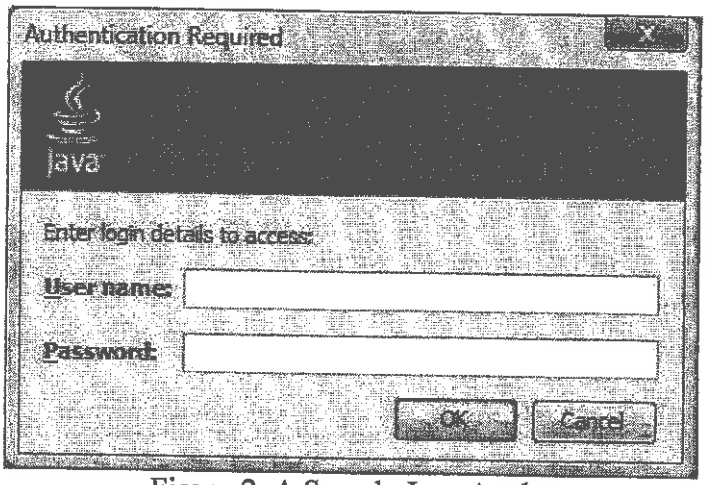

Figure 2. A Sample Java Applet

[3 marks]

Continued...

Question 4

(a) Construct the actionPerformed () function that enables the Java applet to display the string "Object Oriented Programming" when button OOP is pressed and string "Java Programming" when button Java is pressed, as shown in Figure 3 given below.

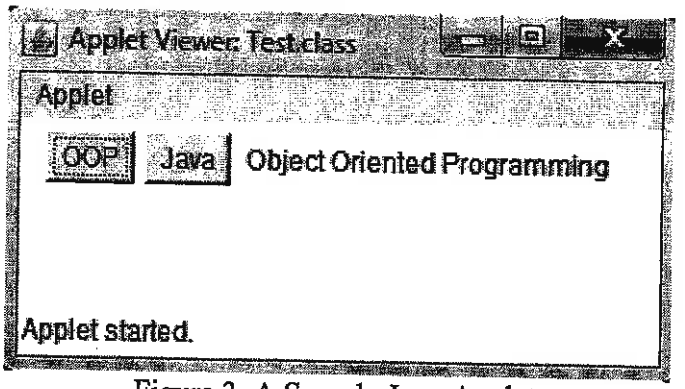

Figure 3. A Sample Java Applet

[4 marks]

```
import java.awt.*;
import java.applet.*;
import java.awt.event.*;

public class Test extends Applet implements ActionListener{
   Button b1=new Button("OOP");
   Button b2=new Button("Java");
   Label l=new Label("Object Oriented Programming");

   public void init(){
     add(b1);
     add(b2);
     add(l);
     b1.addActionListener(this);
     b2.addActionListener(this);
   }
   public void actionPerformed(ActionEvent e){

     // Write your code here




   }
}
```

Continued...

(b) Write a Java program, namely, `FileRead` which takes in file path as the program parameter and displays the file content. You can ignore the exceptions handling. The example output is as depicted in Figure 4.

```
C:\Users\OOP\Desktop>javac FileRead.java

C:\Users\OOP\Desktop>java FileRead C:\Users\OOP\Desktop\sample.txt
Testing 123, OOP file read sample.

C:\Users\OOP\Desktop>
```

Figure 4. Example Output of `FileRead`.

[4 marks]

(c) The program below has a compilation error at the statement marked by the arrow. Write the corrected version of this statement with the exception handling mechanism.

```
import java.io.*;
public class Error{
  public static void main(String[] args){
     FoundError f = new Funny();
     f.getData();  <----------
  }
}

import java.io.*;
class FoundError {
  void getData() throws IOException{
    int i=System.in.read();
    return;
  }
```

[2 marks]

End of Paper